UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/448,164	11/24/1999	PAUL S. GERMSCHEID	33012/277/10	4733

7590 01/26/2004

CHARLES A JOHNSON
UNISYS CORPORATION
LAW DEPARTMENT M S 4773
2470 HIGHCREST ROAD
ROSEVILLE, MN 55113

EXAMINER	
WASSUM, LUKE S	
ART UNIT	**PAPER NUMBER**
2177	15

DATE MAILED: 01/26/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

Technology Center 2100

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
P.O. BOX 1450
ALEXANDRIA, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Paper No. 15

Application Number: 09/448,164
Filing Date: November 24, 1999
Appellant(s): GERMSCHEID ET AL.

John L. Rooney
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 8 December 2003.

(1) Real Party in Interest

A statement identifying the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

A statement identifying the related appeals and interferences which will directly affect or be directly affected by or have a bearing on the decision in the pending appeal is contained in the brief.

(3) Status of Claims

The statement of the status of the claims contained in the brief is correct.

(4) Status of Amendments After Final

No amendment after final has been filed.

(5) Summary of Invention

The summary of invention contained in the brief is correct.

(6) Issues

The appellant's statement of the issues in the brief is correct.

(7) Grouping of Claims

Appellant's brief includes a statement that claims 1-20 do not stand or fall together and provides reasons as set forth in 37 CFR 1.192(c)(7) and (c)(8).

(8) Claims Appealed

The copy of the appealed claims contained in the Appendix to the brief is correct.

(9) Prior Art of Record

6,266,673 HONG et al. 7-2001

UNISYS "Why Do I Need Cool ICE?", Cool ICE and Active Server Pages White Paper, March 1999.

(10) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1-4, 6-8, 11-14 and 16-18 are rejected under 35 U.S.C. § 102(e). Claims 5, 9, 10, 15, 19 and 20 are rejected under 35 U.S.C. § 103(a). These rejections are set forth in prior Office Action, Paper No. 11.

(11) Response to Argument

This Examiner's answer will address the arguments in the order in which they appear in the appeal brief.

A. Issue 1

Claims 1-4, 6-8, 11-14 and 16-18 are properly rejected under 35 U.S.C. § 102(e) as being anticipated by HONG et al.

Regarding Issue 1, Appellants argue that (1) the functionality of **Hong et al.** operates within a single computer, that (2) **Hong et al.** fails to teach the claimed user or Internet terminal, that (3) **Hong et al.** fails to teach the claimed creation of a non-relational empty data set, that (4) the examiner's citation of the **Hong et al.** reference in the rejections of claims 2, 7 and 13 has nothing to do with the claim limitations, and that (5) **Hong et al.** fails to teach the claimed parameter set associated with said non-SQL service request whereby said non-relational empty data set is created in accordance with said parameter set.

Additional arguments against individual claims are presented, but are redundant in light of these five arguments. The examiner's responses to these arguments also address the Appellants' redundant arguments.

In response to these arguments, the Examiner presents the following arguments.

Regarding argument (1), the examiner respectfully submits that the **Hong et al.** reference anticipates the claimed limitation of a publicly accessible digital data network responsively coupled to said Internet terminal and also responsively coupled to said database management system.

In the Background of the Invention, the **Hong et al.** reference discusses a distributed database environment, and how the disclosed invention is particularly useful in such an environment. In the database art, a distributed database is one in which the data is distributed across multiple database management systems running on different nodes on a network. **Hong et al.** explicitly discloses at col. 1, line 64 through col. 2, line 3, the fact that in a distributed database, the client and server are often in a different location.

> "The database to which a client connects is referred to as the local database. Other databases are referred to as remote databases...Often a remote database resides at a location remote to the local database."

Thus, the reference discloses the existence of database clients and servers existing remote from each other, and connected by a network.

Furthermore, the reference teaches the use of the Internet (see item 128 in Figure 1, and also col. 5, lines 40-55), and that "the network link 120 typically provides data communications through one or more networks to other data devices." The reference goes on to teach at col. 5, lines 56-64, that

> "Computer system 100 can send messages and receive data, including program code, through the network(s), network link 120 and communications interface 118. In the Internet example, a server 130 might transmit a requested code for an application program through the Internet 128, ISP 126, local network 122 and communications interface 118. In accordance with the invention, one such downloaded application provides for generation of references to objects in a database as described herein."

In addition, col. 6, lines 53-64 describes the system illustrated in Figure 2, comprising database servers 202 and 252, database server 202 being local to client 208, and database server 252 being remote to client 208.

Finally, the reference teaches that the disclosed system is especially advantageous in a networked environment, where requests are transmitted and objects are returned across a network...Consolidating requests for objects from multiple tables or views reduces network traffic and increases performance.

Given this evidence, the reference teaches the claimed limitation of a publicly accessible digital data network responsively coupled to an Internet terminal and also responsively coupled to a database management system. For example, the examiner interprets Figure 1 as embodying a system wherein a database client runs on computer system 100, and the network link 120 provides access to a database management system running on server 130 via Internet 128. Figure 2 displays a database client 208 running on a server also containing a local database management system 202, and also a remote database management system 252 running on a remote server accessible via a network, for instance, the Internet.

Regarding argument (2), the examiner respectfully submits that the **Hong et al.** reference teaches the claimed Internet terminal.

The Appellants cite the disclosure of their specification in interpreting the 'Internet terminal' limitation. The cited portion discloses "an industry-compatible, personalized computer having a suitable web browser...that communicates over world wide web access using standardized HTML protocol."

First of all, the examiner points out that these details are in the specification and not in the claims. Although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Furthermore, the examiner believes that the hardware overview disclosed in col. 4 anticipates the claimed Internet terminal, since it includes a display 112 with which the user can access the Internet 128 via network link 120.

Regarding argument (3), the examiner respectfully submits that the **Hong et al.** reference teaches the claimed creation of a non-relational empty data set.

The Appellants teach in the specification that the claimed creation of a non-relational data set is the creation of a non-relational table (see replacement page 11, line 10 and lines 19-25; also Figure 14, disclosing the definition of column names and widths for the table).

The **Hong et al.** reference teaches that the client may make requests to create storage organizations (col. 6, lines 14-19), and that storage organizations are tables (see col. 6, lines 44-45), and that the term table is broadly defined and includes, but is not limited to, object tables and relational tables (col. 6, lines 45-48).

The examiner believes that this teaching alone is sufficient to anticipate the claimed creation of a non-relational empty data set. However, consideration of the term 'non-relational' lends further evidence.

The defining characteristic of a relational database is that the records of one table are related to the records of other tables through the use of a common data element, or key. As disclosed in the definition of 'relational database' in the Microsoft Computer Dictionary, "In conducting

searches, a relational database matches information from a field in one table with information in a corresponding field of another table to produce a third table that combines requested data from both tables."

Given such a disclosure, a single table, in isolation, cannot be 'relational', because a table must be related to another table through some key field for it to be relational. Without the ability to link data records from multiple tables through such key fields the term relational cannot apply.

Thus, the examiner believes that even within a relational (or an object-relational) database management system, a non-relational table can be created. Such a table would be non-relational if it were not related to any of the other tables in the database through key fields.

For this additional reason the examiner believes that the **Hong et al.** reference, teaching the ability to create storage organizations/tables, anticipates the claimed creation of a non-relational empty data set.

Regarding argument (4), that the examiner's citation of the **Hong et al.** reference in the rejections of claims 2, 7 and 13 has nothing to do with the claim limitations, the examiner respectfully disagrees.

Claims 2, 7 and 13 add the limitation that the non-relational empty data set of the respective parent claims is stored in a repository. The examiner cited Figure 2, and col. 6, lines 22-64.

Figure 2 shows the existence of databases 204 and 254 (each analogous to the claimed repository), which contain the storage organizations (analogous to the claimed non-relational empty data sets) managed by the database servers 202 and 252 respectively, as disclosed at col. 6, lines 25-27.

Regarding argument (5), that **Hong et al.** fails to teach the claimed parameter set associated with said non-SQL service request whereby said non-relational empty data set is created in accordance with said parameter set, the examiner respectfully disagrees.

As explained above, the **Hong et al.** reference teaches a service request to create a non-relational empty data set at col. 6, lines 14-19 and 44-48. Furthermore, the reference teaches that the service request can be non-SQL (see disclosure that the service requests can be SQL or invoked by software, such as software written in C running on client 208, at col. 6, lines 35-40). Furthermore, **Hong et al.** teaches the claimed parameter set in the form of a data definition, taught beginning at col. 6, line 65, with specific parameter sets illustrated at col. 7, lines 7-12 and 38-40, and also in Figures 3A, 3C, 3E, 3G and 4A. Such data definitions/parameter sets are used by the database management system in the table creation process.

For these reasons, the Examiner maintains that the rejection of the claims 1-4, 6-8, 11-14 and 16-18 is proper, and should be sustained.

B. Issue 2

Claims 5, 9, 10, 15, 19 and 20 are properly rejected under 35 U.S.C. § 103(a) as being anticipated by HONG et al. in view of UNISYS.

Regarding Issue 2, Appellants argue that (1) the combination of the system of **Hong et al.** with MAPPER would not work, because MAPPER is not an object-relational database, and that (2) the examiner has failed to perform the examination required for means-plus-function claims 19 and 20.

Additional arguments against individual claims are presented, but are redundant in light of these five arguments. The examiner's responses to these arguments also address the Appellants' redundant arguments.

In response to these arguments, the Examiner presents the following arguments.

Regarding argument (1), the examiner respectfully submits that although the **Hong et al.** reference teaches a system relating to 'object-relational databases', the database functionality taught and utilized in the disclosure would have been supported by a relational database management system, such as MAPPER. For instance, **Hong et al.** teaches that for the purposes of the disclosed system, objects are analogous to records in tables ("...an object table contains objects belonging to the same object type, i.e., the table type.", col. 7, lines 49-50). Another example would be the use of row-ids in generating references (beginning at col. 10, line 60), row-ids being a feature supported by relational database management systems.

Absent some specific teaching as to why the system of **Hong et al.** would be incompatible with a relational database management system, the examiner disagrees with the broad argument advanced by the Appellants.

Regarding argument (2), the examiner respectfully disagrees.

In the rejection of claim 19, the Appellants further limit the "offering means". This offering means was addressed in the rejection of parent claim 16, in paragraph 9 of the Final Rejection of record, paper number 11.

The new limitation is that the offering means comprises the MAPPER database management system. In rejecting the claim, the examiner has made reference to the rejection of the parent claim, while noting that in the parent claim, the database management system as disclosed is not MAPPER.

The examiner then proceeds to explain how the **UNISYS** reference teaches the MAPPER database management system, and reasons why it would have been obvious to one of ordinary skill in the art at the time of the invention to incorporate the MAPPER database into the system taught by the **Hong et al.** reference.

The examiner believes that the rejection as crafted and structured is proper with regards to the rejected claim's status as means-plus-function.

Analogously, claim 20 further limits the claimed "permitting means", which was addressed in the rejection of parent claim 16, in paragraph 9 of the Final Rejection of record, paper number 11.

Once again, the examiner believes that the rejection as crafted and structured is proper with regards to the rejected claim's status as means-plus-function.

Furthermore, the cited portions of the **Hong et al.** reference (col. 4, lines 7-38 and col. 5, lines 25-64) have direct bearing on the fact that the claimed permitting means is an industry standard personal computer. While the reference might fail to explicitly use the phrase "industry standard personal computer", the examiner finds it difficult to comprehend how the Appellants can see the cited portions of the reference as "irrelevant". The fact that the claimed personal computer is "industry standard" means that in the absence of any teaching to the contrary, a personal computer should be assumed to be "industry standard". The cited portions of the **Hong et al.** reference disclose the component parts of a computer system that is fully consistent with an "industry standard personal computer".

For these reasons, the Examiner maintains that the rejection of claims 5, 9, 10, 15, 19 and 20 is proper, and should be sustained.

Conclusion

Claims 1-4, 6-8, 11-14 and 16-18 are properly rejected under 35 U.S.C. § 102(e), and claims 5, 9, 10, 15, 19 and 20 are properly rejected under 35 U.S.C. § 103(a).

In light of the foregoing arguments, the Examiner respectfully requests the Honorable Board of Appeals to sustain the rejections.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Luke S. Wassum
Art Unit 2177

Conferees
Srirama T. Channavajjala
Jean R. Homere

lsw
January 20, 2004

John E. Breene

JOHN BREENE
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100

CHARLES A JOHNSON
UNISYS CORPORATION
LAW DEPARTMENT M S 4773
2470 HIGHCREST ROAD
ROSEVILLE MN 55113

Microsoft Press
Computer Dictionary

Third Edition

Microsoft Press

PUBLISHED BY
Microsoft Press
A Division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052-6399

Library of Congress Cataloging-in-Publication Data
Microsoft Press Computer Dictionary. -- 3rd ed.
p. cm.
ISBN 1-57231-446-X
1. Computers--Dictionaries. 2. Microcomputers--Dictionaries.
I. Microsoft Press.
QA76.15.M54 1997
004'.03--dc21 97-15489
CIP

Printed and bound in the United States of America.

2 3 4 5 6 7 8 9 QMQM 2 1 0 9 8 7

Distributed to the book trade in Canada by Macmillan of Canada, a division of Canada Publishing Corporation.

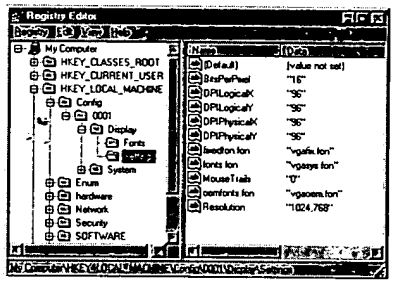
A CIP catalogue record for this book is available from the British Library.

Microsoft Press books are available through booksellers and distributors worldwide. For further information about international editions, contact your local Microsoft Corporation office. Or contact Microsoft Press International directly at fax (425) 936-7329.

Acquisitions Editor: Kim Fryer
Project Editor: Maureen Williams Zimmerman, Anne Taussig
Technical Editors: Dail Magee Jr., Gary Nelson, Jean Ross, Jim Fuchs, John Conrow, Kurt Meyer, Robert Lyon, Roslyn Lutsch

replaces most of the text-based .ini files used in Windows 3.*x* and MS-DOS configuration files, such as AUTOEXEC.BAT and CONFIG.SYS. Although the Windows 95 Registry is similar to the one in Windows NT, there are some differences, such as how they are stored on disk. *Also called* System Registry. *See also* hierarchical database, .ini, input/output port, property sheet, Registry Editor.

Registry Editor \rej´is-trē ed`i-tər\ *n.* An application under Windows 95 that allows the user to edit the entries in the Registry. See the illustration. *Acronym:* REGEDIT (rej´ed`it). *See also* Registry.



Registry Editor.

regression analysis \ri-gresh´ən ə-nal`ə-sis\ *n.* In statistics, an analysis of the degree to which variations in an independent variable affect a dependent variable (a variable whose value depends on the value of another variable). *See also* multiple regression.

regression testing \ri-gresh´ən te`stēng\ *n.* Complete retesting of a modified program, rather than a test of only the modified routines, to ensure that no errors have been introduced with the modifications.

relation \rə-lā´shən\ *n.* A structure composed of attributes (individual characteristics, such as name or address, corresponding to the columns in a table) and tuples (sets of attribute values describing particular entities, such as customers, corresponding to the rows in a table). Within a relation, tuples cannot be repeated; each must be unique. Further, tuples are unordered within a relation; interchanging two tuples does not change the relation. Finally, if relational theory is to be applicable, the domain of each attribute must be atomic—that is, a simple value, rather than a structure such as an array or a record. A relation in which the domains of all attributes are atomic is said to be normalized or in first normal form. *See also* normal form (definition 1).

relational algebra \rə-lā`shən-əl al´jə-brə\ *n.* A collection of rules and operators that permit relations (tables) to be manipulated. Relational algebra is usually described as having the following operators: SELECT, PROJECT, PRODUCT, UNION, INTERSECT, DIFFERENCE, JOIN (or INNER JOIN), and DIVIDE. In a relational database, relational algebra is used to develop procedures to build new relations based on the existing relations.

relational calculus \rə-lā`shən-əl kal´kyə-lus\ *n.* In database management, a nonprocedural method for manipulating relations (tables). There are two families of relational calculus: domain calculus and tuple calculus. The two families of relational calculus are mathematically equivalent to each other and to relational algebra. Using either family, one can formulate a description of a desired relation, based on the existing relations in the database.

relational database \rə-lā`shən-əl dā´tə-bās\ *n.* A database or database management system that stores information in tables—rows and columns of data—and conducts searches by using data in specified columns of one table to find additional data in another table. In a relational database, the rows of a table represent records (collections of information about separate items) and the columns represent fields (particular attributes of a record). In conducting searches, a relational database matches information from a field in one table with information in a corresponding field of another table to produce a third table that combines requested data from both tables. For example, if one table contains the fields EMPLOYEE-ID, LAST-NAME, FIRST-NAME, and HIRE-DATE, and another contains the fields DEPT, EMPLOYEE-ID, and SALARY, a relational database can match the EMPLOYEE-ID fields in the two tables to find such information as the names of all employees earning a certain salary or the departments of all employees hired after a certain date. In other words, a

relational database uses matching values in two tables to relate information in one to information in the other. Microcomputer database products typically are relational databases. *Compare* flat-file database, inverted-list database.

relational database management system \rə-lā`shən-əl dā`tə-bās man´əj-mənt si`stəm\ *n.* *See* relational database.

relational expression \rə-lā`shən-əl eks-presh´ən\ *n.* An expression that uses a relational operator such as "less than" or "greater than" to compare two or more expressions. A relational expression resolves to a Boolean (true/false) value. *See also* Boolean, relational operator.

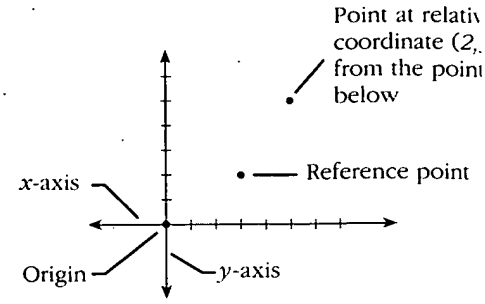
relational model \rə-lā`shən-əl mod´əl\ *n.* A data model in which the data is organized in relations (tables). This is the model implemented in most modern database management systems.

relational operator \rə-lā`shən-əl op´ər-ā-tər\ *n.* An operator that allows the programmer to compare two (or more) values or expressions. Typical relational operators are greater than (>), equal to (=), less than (<), not equal to (<>), greater than or equal to (>=), and less than or equal to (<=). *See also* relational expression.

relational structure \rə-lā`shən-əl struk´chur\ *n.* The record organization used in the implementation of a relational model.

relative address \rel`ə-tiv a´dres, ə-dres´\ *n.* A location, as in a computer's memory, that is specified in terms of its distance (displacement or offset) from a starting point (base address). A relative address is typically computed by adding an offset to the base—in everyday terms, this is similar to creating the address 2001 Main Street, in which the base is the 2000 block of Main Street and the offset is 1, which specifies the first house from the beginning of the block. *Also called* indirect address.

relative coordinates \rel`ə-tiv kō-ōr´də-nəts\ *n.* Coordinates that are defined in terms of their distance from a given starting point, rather than from the origin (intersection of two axes). For example, from a starting point on the screen, a square defined by relative coordinates can be drawn as a series of lines, each representing a displacement in distance and direction from the end of the preceding point. The entire square can be redrawn at another location simply by changing the coordinates of the starting point rather than by recalculating the coordinates of each corner with reference to the origin. See the illustration. *Compare* absolute coordinates.



Relative coordinates.

relative movement \rel`ə-tiv mo͞ov´mənt\ **1.** Motion whose distance and direction are r tive to a starting point. For example, whe mouse pointer is moved on the screen, the c dinates of its new position are relative to the vious location of the pointer. *See also* rel coordinates, relative pointing device. **2.** In c puter graphics and cinematography, the m ment of one object in relation to another, suc the movement of horse A from the perspectiv horse B on a racetrack.

relative path \rel`ə-tiv path´\ *n.* A path th implied by the current working directory. Wh user enters a command that refers to a file, i full pathname is not entered, the current wor directory becomes the relative path of the referred to. *Compare* full path.

relative pointing device \rel`ə-tiv poin´tĕng vīs\ *n.* A cursor-control device, such as a m or a trackball, in which the movement of an screen cursor is linked to the movement of device but not to the position of the device. example, if a user picks up a mouse and pu down in a different location on a desk, the p tion of the on-screen cursor does not ch because no movement (rolling) is detected. W the user rolls the mouse again, the cursor m to reflect the mouse movement against the su of the desk. Relative pointing devices differ absolute pointing devices, such as graphics lets, in which the device's location with